Amendments to the Claims:

This listing of claims replaces all prior versions and listings of claims in the application:

Listing of Claims:

1. (**Currently Amended**) A computer program product, tangibly embodied in a machine readable storage device, the computer program product comprising instructions operable to cause data processing apparatus to perform operations comprising:

receiving a value from a user of a client computer through a first user interface (UI) element of **a** displayed page, wherein a document object model (DOM) hierarchy locally stored on the client computer defines attributes of a plurality of UI elements, including the first UI element, rendered through a Web browser, at least a child node of the DOM hierarchy defines one or more attributes of the first UI element;

in response to at least receiving the value, identifying in the Web browser in the client computer an update to the one or more attributes of the first UI element, wherein the update modifies presentation of the first UI element;

determining in the browser whether the update to the one or more attributes of the first UI element can be applied to the locally-stored DOM hierarchy independent of receiving an update to the hierarchy from a server;

if the change can be locally applied, automatically modifying the child node in the locally-stored DOM hierarchy to dynamically update a presentation of the first UI element without re-rendering all of the displayed page; and

else, setting a dirty flag.

2. **(Cancelled)**

3. **(Original)** The product of claim 1, wherein the dirty flag indicates invalidity of a cached DOM representation of the UI element.

4. **(Original)** The product of claim 1, wherein the dirty flag indicates invalidity of a cached DOM representation of a further UI element that is a parent of the UI element.

5. (**Previously Presented**) The product of claim 3, wherein the cached DOM representation comprises the child node and the one or more attributes are an attribute of the child node.

6. **(Previously Presented)** The product of claim 4, wherein the cached DOM representation comprises the child node and the one or more attributes are an attribute of the child node.

7. **(Previously Presented)** The product of claim 3, wherein the cached DOM representation comprises a sub hierarchy of the DOM hierarchy and the child node is the root node of the sub hierarchy and the one or more attributes are an attribute of a sub node of the sub hierarchy.

8. **(Previously Presented)** The product of claim 4, wherein the cached DOM representation comprises a sub hierarchy of the DOM hierarchy and the child node is the root node of the sub hierarchy and the one or more attributes are an attribute of a sub node of the sub hierarchy.

9. **(Previously Presented)** A computer program product, tangibly embodied in a machine readable storage device, the computer program product comprising instructions operable to cause data processing apparatus to perform operations comprising:

receiving a request to render a user interface (UI) element of a UI tree in a client computer, the UI tree representing in the client computer a hierarchical structure of UI elements contained in a user interface page received in the client computer from a server computer, the UI element being an element of the user interface page;

determining whether a dirty flag is set for a cached document object model (DOM) representation of a flagged UI element that is on the path of the UI element in the UI tree, the flagged UI element having a reference to a node of the DOM hierarchy, the cached DOM representation being in the client computer, the UI element being different from the flagged UI element;

if the dirty flag is set, generating in the client computer a new DOM representation of the flagged UI element and inserting the new DOM representation into the DOM hierarchy by using the UI tree and the reference to the node; and

else, inserting, independent of receiving an update to the DOM hierarchy from a server, the cached DOM representation in the client computer into the DOM hierarchy by using the UI tree and the reference to the node to dynamically update a presentation of the UI element without re-rendering all of the user interface page.

10. **(Original)** The product of claim 9, wherein the cached DOM representation is a cached sub tree of the DOM hierarchy and the node is the root node of the cached sub tree.

11. **(Original)** The product of claim 9, wherein the new DOM representation is a new sub tree of the DOM hierarchy and the node is the root node of the new sub tree.

12. **(Currently Amended)** Apparatus comprising:

means for receiving a value from a user of a client computer through a first user interface (UI) element of **a** displayed page, wherein a document object model (DOM) hierarchy locally stored on the client computer defines attributes of a plurality of UI elements, including the first UI element, rendered through a Web browser, at least a child node of the DOM hierarchy defines one or more attributes of the first UI element;

in response to at least receiving the value, means for identifying in the Web browser in the client computer an update to the one or more attributes of the first UI element, wherein the update modifies presentation of the first UI element;

means for determining in the browser whether the update to the one or more attributes of the first UI element can be applied to the locally-stored DOM hierarchy independent of receiving an update to the hierarchy from a server;

means for, if the change can be locally applied, automatically modifying the child node in the locally-stored DOM hierarchy to dynamically update a presentation of the first UI element without re-rendering all of the displayed page; and

else, setting a dirty flag.

13. **(Cancelled)**

14. **(Original)** The apparatus of claim 12, wherein the dirty flag indicates invalidity of a cached DOM representation of the UI element.

15. **(Previously Presented)** Apparatus comprising:

means for receiving a request to render a user interface (UI) element of a UI tree in a client computer, the UI tree representing in the client computer a hierarchical structure of UI elements contained in a user interface page received in the client computer from a server computer, the UI element being an element of the user interface page;

determining whether a dirty flag is set for a cached document object model (DOM) representation of a flagged UI element that is on the path of the UI element in the UI tree; the flagged UI element having a reference to a node of the DOM hierarchy, the cached DOM representation being in the client computer, the UI element being different from the flagged UI element; and

means for, if the dirty flag is set, generating in the client computer a new DOM representation of the flagged UI element and inserting the new DOM representation into the DOM hierarchy by using the UI tree and the reference to the node; and

else, inserting**,** independent of receiving an update to the DOM hierarchy from a server, the cached DOM representation in the client computer into the DOM hierarchy by using the UI tree and the reference to the node to dynamically update a presentation of the UI element without re-rendering all of the user interface page.